

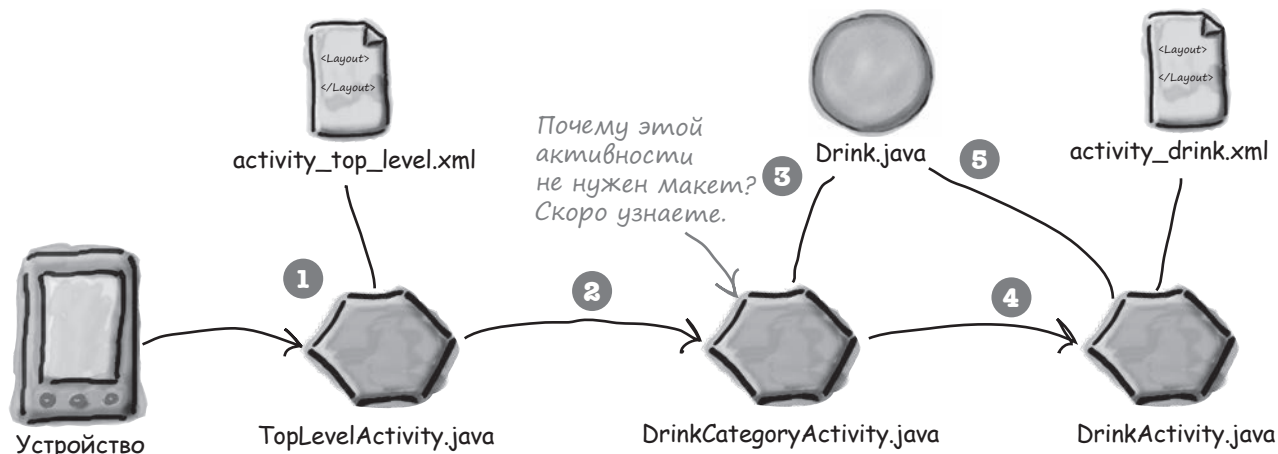
Структура приложения Starbuzz

Приложение состоит из трех активностей. `TopLevelActivity` — активность верхнего уровня — обеспечивает основную навигацию по разделам приложения. `DrinkCategoryActivity` — активность категории со списком напитков. Третья активность, `DrinkActivity`, содержит подробную информацию о конкретном напитке.

В этой версии данные напитков будут храниться в классе Java. В одной из следующих глав информация будет перенесена в базу данных, но пока мы хотим сосредоточиться на построении приложения, не отвлекаясь на поддержку баз данных.

- 1 **При запуске приложения открывается активность `TopLevelActivity`.**
Активность использует макет `activity_top_level.xml`. Активность выводит список команд для перехода к разделам напитков, блюд и кофеен.
- 2 **Пользователь выбирает команду Drinks в `TopLevelActivity`.**
Команда открывает активность `DrinkCategoryActivity`, которая отображает список напитков.
- 3 **Информация о напитках хранится в файле класса `Drink.java`.**
`DrinkCategoryActivity` получает данные напитков из этого класса.
- 4 **Пользователь выбирает напиток в `DrinkCategoryActivity`.**
При выборе запускается активность `DrinkActivity`, использующая макет `activity_drink.xml`.
- 5 **`DrinkActivity` получает подробную информацию о напитке из файла класса `Drink.java`.**

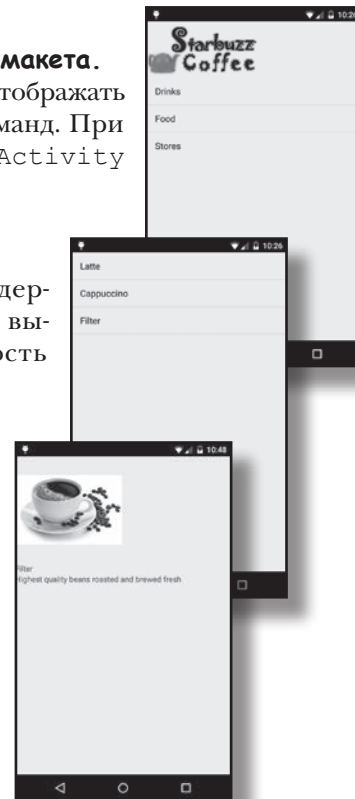
Для активности `DrinkCategoryActivity` создавать макет не нужно — вскоре вы увидите, почему.



Последовательность действий

Ниже перечислены основные этапы построения приложения:

- 1 **Добавление класса Drink и ресурсов изображений.**
Класс содержит подробную информацию о напитках; также в приложении используются ресурсы изображений напитков и логотипа Starbuzz.
- 2 **Создание активности TopLevelActivity и ее макета.**
«Точка входа» приложения: активность должна отображать логотип Starbuzz и навигационный список команд. При выборе команды Drink активность TopLevelActivity должна открывать DrinkCategoryActivity.
- 3 **Создание DrinkCategoryActivity.**
Активность DrinkCategoryActivity содержит список всех имеющихся напитков. При выборе напитка должна открываться активность DrinkCategory.
- 4 **Создание активности DrinkActivity и ее макета.**
Активность DrinkActivity выводит информацию о напитке, выбранном пользователем в списке DrinkCategoryActivity.



Создание проекта

Проект приложения создается точно так же, как это делалось в предыдущих главах.

Создайте новый проект Android для приложения с именем “Starbuzz” и именем пакета `com.hfad.starbuzz`. Выберите минимальный уровень SDK равным API 15. Приложение должно содержать активность с именем “TopLevelActivity” и макет с именем “activity_top_level”.



Добавление ресурсов
TopLevelActivity
DrinkCategoryActivity
DrinkActivity

Класс Drink

Для начала добавим в приложение класс Drink. *Drink.java* — обычный файл класса Java, из которого активности будут получать данные напитков. Класс определяет массив из трех объектов, представляющих напитки; каждый объект состоит из названия, описания и идентификатора ресурса изображения. Добавьте класс в пакет *com.hfad.starbuzz* в папке *app/src/main/java* вашего проекта, присвоив ему имя Drink. Сохраните изменения.



Добавление ресурсов
TopLevelActivity
DrinkCategoryActivity
DrinkActivity

```
package com.hfad.starbuzz;

public class Drink {
    private String name;
    private String description;
    private int imageResourceId;

    //drinks — массив из трех объектов Drink
    public static final Drink[] drinks = {
        new Drink("Latte", "A couple of espresso shots with steamed milk",
            R.drawable.latte),
        new Drink("Cappuccino", "Espresso, hot milk, and a steamed milk foam",
            R.drawable.cappuccino),
        new Drink("Filter", "Highest quality beans roasted and brewed fresh",
            R.drawable.filter)
    };

    //Для каждого напитка хранится имя, описание и ресурс изображения
    private Drink(String name, String description, int imageResourceId) {
        this.name = name;
        this.description = description;
        this.imageResourceId = imageResourceId;
    }

    public String getDescription() {
        return description;
    }

    public String getName() {
        return name;
    }

    public int getImageResourceId() {
        return imageResourceId;
    }

    public String toString() {
        return this.name;
    }
}
```

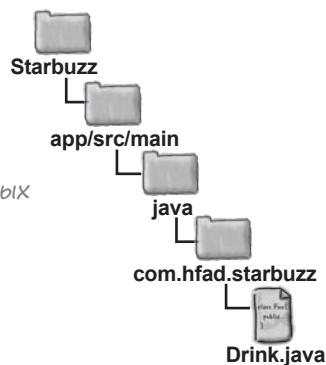
Каждый объект Drink состоит из полей имени, описания и идентификатора ресурса изображения. Идентификаторы ресурсов принадлежат изображениям напитков, которые будут добавлены в проект на следующей странице.

Изображения напитков. Мы добавим их на следующем этапе.

Конструктор Drink.

Get-методы для приватных переменных.

В качестве строкового представления Drink используется название напитка.

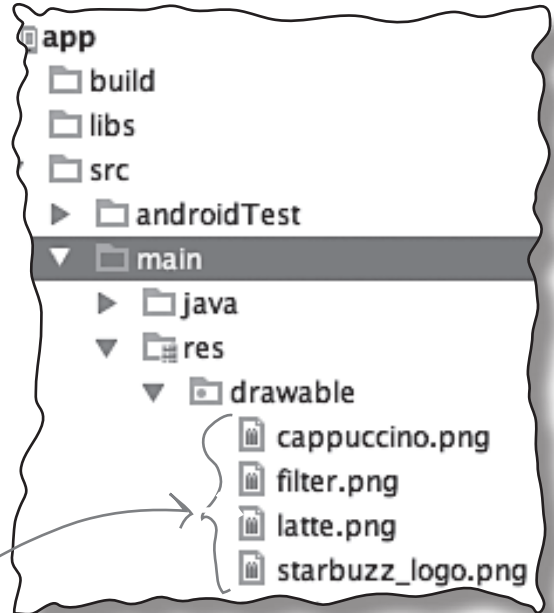


Файлы изображений

Код `Drink` включает три ресурса изображений напитков с идентификаторами `R.drawable.latte`, `R.drawable.cappuccino` и `R.drawable.filter`. Они нужны для того, чтобы пользователь мог увидеть фотографию напитка. `R.drawable.latte` ссылается на файл изображения с именем *latte*, `R.drawable.cappuccino` — на файл изображения с именем *cappuccino*, а `R.drawable.filter` — на файл изображения с именем *filter*.

Эти файлы изображений необходимо добавить в проект вместе с изображением логотипа *Starbuzz*, чтобы его можно было использовать в активности верхнего уровня. Для этого загрузите файлы *starbuzz-logo.png*, *cappuccino.png*, *filter.png* и *latte.png* по адресу <https://tinyurl.com/HeadFirstAndroid> и перетащите их в папку `app/src/main/res/drawable` в проекте *Starbuzz*. При добавлении изображений в приложение необходимо решить, собираетесь ли вы использовать разные изображения для экранов с разной плотностью пикселей. В нашем примере одно изображение будет использоваться для всех экранов, поэтому достаточно поместить один экземпляр изображения в одну папку. Если вы в своем приложении решите использовать разные версии графики для разных экранов, разместите разные варианты изображений в папках *drawable**, как описано в главе 5.

Четыре файла изображений. Добавьте их в Android Studio перетаскиванием в папку *drawable*.



При сохранении изображений в проекте Android назначает им идентификаторы в формате `R.drawable.имя_файла`. Например, изображению из файла *latte.png* присваивается идентификатор `R.drawable.latte`, соответствующий значению ресурса изображения *latte* из класса `Drink`.



Drink

name: "Latte"
description: "A couple of
expresso shots with steamed
milk"
imageResourceId: R.drawable.latte

Изображению *latte.png* присваивается идентификатор `R.drawable.latte`.



R.drawable.latte

После добавления класса `Drink` и ресурсов изображений в проект можно переходить к активностям. Начнем с активности верхнего уровня.